HUMBOLDT-UNIVERSITÄT ZU BERLIN

LEBENSWISSENSCHAFTLICHE FAKULTÄT
INSTITUT FÜR BIOLOGIE

MASTERARBEIT

ZUM ERWERB DES AKADEMISCHEN GRADES
MASTER OF SCIENCE

Dynophore:
Dynamische Pharmakophore

Implementierung der Generierung von Pharmakophoren
auf der Basis von Moleküldynamik-Trajektorien
und ihre graphische Darstellung

Dynophores:
Novel Dynamic Pharmacophores

Implementation of Pharmacophore Generation
Based on Molecular Dynamics Trajectories
and Their Graphical Representation

vorgelegt von
Dominique Sydow

angefertigt in der Arbeitsgruppe
Computergestütztes Wirkstoffdesign
Pharmazeutische und Medizinische Chemie
Institut für Pharmazie
Freie Universität Berlin

30. November 2015

Supervision

Internal Supervision

Prof. Edda Klipp
Theoretische Biophysik
Institut für Biologie
Humboldt-Universität zu Berlin

External Supervision

Prof. Gerhard Wolber
Computergestütztes Wirkstoffdesign
Institut für Pharmazie
Freie Universität Berlin

Acknowledgements

First of all, I want to thank Prof. Gerhard Wolber for welcoming me in his group at the Freie Universität Berlin, and giving me the chance to combine Java programming with the interesting field of computer-aided drug design. I am very grateful for his structured guidance and friendly support during my project.

I acknowledge Christin Rakers and Dr. Marcel Bermudez for giving me the opportunity to test my program with molecular dynamics simulations for their projects SULT1E1 and M_2AChR, respectively. Discussions about the results and potential applications constantly improved my work.

I thank the whole group of Prof. Wolber for their enthusiasm about dynophores, and their suggestions and discussions on my project.

Prof. Gerhard Wolber, Christin Rakers, Dr. Marcel Bermudez and Dr. Jérémie Mortier improved my written master thesis by valuable suggestions and advice.

Furthermore, I want to thank the Frauenförderung of the Fachbereich Biologie, Chemie, Pharmazie at the Freie Universität Berlin for supporting my visit at the Summer School Drug Design in Vienna. This conference was a great opportunity for me to learn a lot about techniques in drug design, and to present and discuss my work outside of my group.

I am obliged to the Friedrich-Ebert-Stiftung for their promotion and financial support during my Bachelor and Master studies.

I am thankful to Ferdinand Krupp for his support and confidence during my master studies, as well as his never-ending enthusiasm for scientific discussions.

Last but not least, I am very grateful to Prof. Edda Klipp for her supervision and evaluation of my master thesis on behalf of the Humboldt-Universität zu Berlin.

Abstract

In medicinal chemistry and drug discovery, pharmacophore modelling has become a well established method for molecular design. A 3D pharmacophore model represents the ensemble of universal steric and chemical features that stand for a specific mode of binding. Currently, these pharmacophores are created from a multi-conformational ligand set (ligand-based drug design) or from a ligand-target complex (structure-based drug design). However, classical pharmacophore models only allow a static view on a single ligand-target conformation. The aim of the presented study was to develop pharmacophore models that represent conformational flexibility derived from molecular dynamics (MD) simulations. Therefore, pharmacophore features from the MD trajectory frames are grouped into so-called *superfeatures* by their feature type and involved atoms on ligand-site. Subsequently, these superfeatures and their environmental interaction partners are statistically and sequentially analysed in terms of their occurrence and interaction distance. These dynamic pharmacophores are termed *dynophores*, and graphically represented as feature point clouds for superfeature space analysis. Dynophores potentially enhance the predictive power of 3D pharmacophores, and serve as powerful analysis and validation tool for the evaluation of pharmacophores, ligand-target interaction patterns and MD simulations. Dynophore generation and analysis were implemented as *DynophoreApp* program within the ilib/LigandScout framework, a Java-based software for *in silico* drug discovery.

Zusammenfassung

In der medizinischen Chemie und Wirkstoffforschung haben sich Pharmakophormodelle als wichtige Methode für molekulares Design etabliert. Ein 3D Pharmakophormodell repräsentiert ein Ensemble von universellen sterischen und chemischen Eigenschaften (Features), die einen spezifischen Bindungsmodus beschreiben. Diese Pharmakophore werden auf der Grundlage eines Sets von Liganden in verschiedenen Konformationen (Liganden-basiertes Wirkstoffdesign) oder aus einem Liganden-Target-Komplex (Struktur-basiertes Wirkstoffdesign) erstellt. Allerdings erlauben klassische Pharmakophore lediglich eine statische Sicht auf eine einzige Liganden-Target-Konformation. Ziel dieser Arbeit war die Entwicklung eines Pharmakophormodells, das die Konformationsflexibilität aus Molekül̈dynamik (MD)-Simulationen abbildet. Dafür werden die Pharmakophor-Features für jeden Trajektorienzeitschritt in so genannte *Superfeatures* gruppiert. Die Gruppierung erfolgt anhand des Feature-Typs und der Ligandenatome, welche an dem Feature beteiligt sind. Nachfolgend werden die Superfeatures und ihre Interaktionspartner statistisch und sequentiell hinsichtlich ihres Auftretens und dem Abstand zwischen den Interaktionspartnern ausgewertet. Diese dynamischen Pharmakophore werden *Dynophore* genannt und graphisch als Feature-Punktwolke für eine räumlichen Analyse der Superfeatures dargestellt. Dynophore haben das Potential, die Vorhersagekraft von 3D Pharmakophoren zu verbessern und dienen gleichzeitig als Analyse- und Validierungswerkzeug zur Bewertung von Pharmakophoren, Liganden-Target-Interaktionsmustern und MD-Simulationen. Das Programm *DynophoreApp* wurde zur Generierung und Analyse von Dynophoren im Rahmen des ilib/LigandScout-Frameworks implementiert, einer Java-basierten Software für *in silico* Wirkstoffdesign.

Contents

CONTENTS

List of Figures

List of Tables

Acronyms

4-OH-HCB	4-OH-2,3,5,2',4',5'-hexachlorobiphenyl (SULT1E1 ligand)
AR	aromatic ring (LigandScout chemical feature)
Asp	asparagine
CADD	computer-aided drug discovery/design
DCD	binary MD data file
GPCR	G protein coupled receptor
H	hydrophobic area (LigandScout chemical feature)
HBA	hydrogen bond acceptor (LigandScout chemical feature)
HBD	hydrogen bond donor (LigandScout chemical feature)
His	histidine
M_2AChR	muscarinic M_2 acetylcholine receptor
MD	molecular dynamics
NI	negative ionisable (LigandScout chemical feature)
PAPS	phosphoadenosine-5'-phosphosulfonate (SULT1E1 cofactor)
PDB	Protein Data Base
PI	positive ionisable (LigandScout chemical feature)
PML	pharmaceutical markup language
PMZ	compressed pharmaceutical markup language
RMSD	root mean square deviation
SULT1E1	sulfotransferase subtype 1E1
VMD	Visual Molecular Dynamics

1 Introduction

1.1 Computer-Aided Drug Design

Computer-aided drug discovery (CADD) has evolved to an important method during the development of therapeutically relevant small molecules. Due to its more targeted and rational approach when compared to traditional high-throughput screening and combinatorial chemistry, CADD aims to significantly decrease the number of compounds necessary for experimental screening while increasing the hit rate of novel drug molecules. It not only intends to explain the molecular basis of therapeutic activity, but also to predict possible derivatives that would improve activity. In the drug discovery pipeline (Figure 1), computational methods are usually used to (i) filter predicted active compounds from large compound libraries for experimental testing, (ii) optimise lead compounds by increasing its affinity or improve properties including absorption, distribution, metabolism, excretion, and toxicity (ADMET), and (iii) design novel compounds [1].

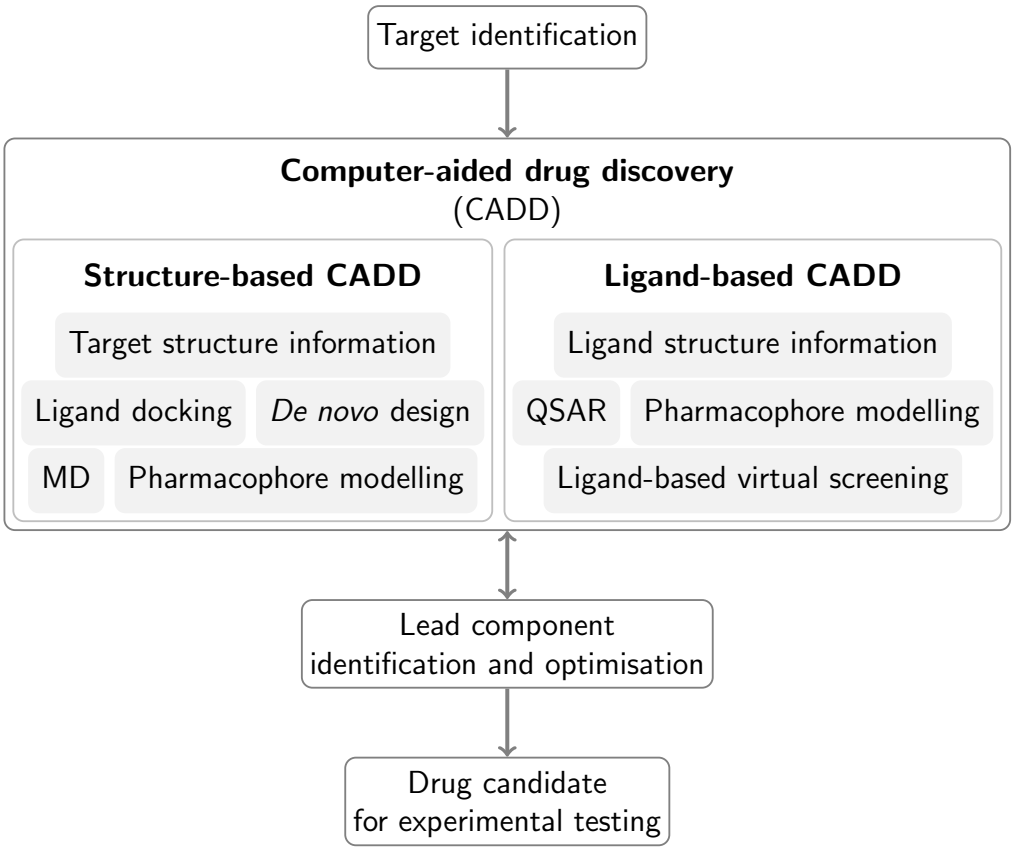


Figure 1: **Computer-aided drug discovery (CADD) pipeline**: A therapeutically relevant target is identified, and a structure- or ligand-based approach is selected for drug discovery, depending on the availability of structural information. After successful identification of multiple lead compounds, several cycles of lead optimisation result in one or more drug candidates [1].

A target of therapeutic interest is selected and enters the CADD pipeline (Figure 1): Depending on the availability of structural information on target and ligand, a structure- or ligand-based approach is selected for drug discovery. In ligand-based drug design on the one hand, information about ligand structures and their associated activity can be described using quantitative structure-activity relationship (QSAR) models or summarised using pharmacophore models for ligand-based virtual screening. In structure-based drug design on the other hand, information about the target structure is used for ligand docking, *de novo* design of ligands, molecular dynamics (MD) studies and/or pharmacophore modelling of ligands bound to a target. After successful identification of bio-active compounds, several cycles of lead optimisation result in one or more drug candidates.

In the study presented, structure-based techniques of pharmacophore modelling and MD studies are combined to an extended pharmacophore model including previously missing information on ligand-target dynamics and conformational states.

1.2 Ligand-Target Interaction Mapping Using Molecular Dynamics

Macromolecular structures are mainly explored using X-ray diffraction of crystallised proteins. However, this technique only provides a static conformation, one snapshot of multiple possible conformations. Flexibility of a molecular system can be derived from nuclear magnetic resonance (NMR) experiments, which show an ensemble of conformations of a system, but which are also very challenging to interpret for larger systems. Hence, complementary tools for the investigation of target and ligand binding dynamics are needed. *In silico* MD simulations can be used to investigate conformational dynamics of a protein-ligand complex. Such simulations can yield dynamic information of a system on an atomic level, not only in terms of protein dynamics and its configuration states but also in terms of target-ligand interactions. Thus, MD has emerged as an important technique for drug design [2].

Several different MD-based approaches for protein-ligand interaction mapping are described in literature and conceptually summarised here:

(i) *Multiple-copy simultaneous search (MCSS)* generates functionality maps, which describe favourable ligand binding derived from MD simulations of functional fragments that are simultaneously and randomly placed on the protein surface [3].

(ii) In the *locally enhanced sampling (LES)* method, ligand conformations are sampled from a simulation with several ligand copies that do not interact with each other but do interact with the environment [4].

(iii) The *relaxed complex method (RSC)* docks whole ligands to an ensemble of target conformations derived from MD simulations with subsequent rescoring [5, 6, 7].

(iv) The *lambda dynamics method* is another multiple-copy method for speeding up free-energy calculations. Multiple ligands are simultaneously located in the receptor

binding site, and the barriers for conformational transitions are lowered. This allows each ligand to further explore orientational and conformational space [8].

(v) Structure-based pharmacophore models represent the ensemble of universal steric and chemical features of a ligand within its structural environment. In *MD-based static pharmacophore modelling*, also referred to as dynamic pharmacophore modelling, a pharmacophore model is created from several ligand-target conformations derived from MD simulations. Such approaches will be discussed in the following in more detail and form the starting point of this study.

MD-based static pharmacophores. In literature, several approaches include MD simulations in the generation of static pharmacophore models. These approaches are generally composed of three steps:

1. Select structural conformations from MD trajectories: MD simulations of the target with or without ligand are conducted, and representative conformations are generated with different clustering methods.

2. Define pharmacophore models for each conformation: Pharmacophore features for each selected conformational snapshot are defined with various tools and methods. These methods include simulating probe molecule positions at the active site [9], decision-based positioning of probes in box grids covering the active site [10], or the use of Metropolis configurational sampling for fixed target and probe molecules [11]. Those methods were used to create MD-based static pharmacophores for HIV-1 integrase [9, 10, 11]. Other probing techniques use LUDI interaction maps [12], or molecular interaction field analysis [13]. Furthermore, docking can serve as basis for pharmacophore modelling, such as probe docking followed by energy mapping, clustering and ranking [14, 15], or ligand screening against all selected conformational snapshots using ensemble docking [16, 17] with subsequent pharmacophore generation for docked ligand hits.

3. Superpose all pharmacophore models, and decide on a generalised pharmacophore model: An MD-based static pharmacophore model is generated by superposition of all pharmacophores per conformational snapshot and rule-based selection of common features.

However, these MD-based static pharmacophore models represent a selection of conformational states but not the dynamic nature of the MD trajectory. Thus, the aim of this work was to include MD-based information on flexibility of a system in a dynamic pharmacophore modelling approach.

1.3 Aim of This Work

This work aimed at including conformational and dynamic information derived from MD simulations in one single dynamic pharmacophore model. Therefore, an automated workflow for integrated dynamic pharmacophore generation, analysis and representation was implemented using the application programming interface (API) of the ilib/LigandScout framework. This novel dynamic pharmacophore is termed *dynophore* and the associated program *DynophoreApp*. This approach allows the usage of whole MD trajectories or user-defined trajectory intervals, and summarises LigandScout three-dimensional (3D) pharmacophores for each trajectory snapshot in one comprehensive dynophore. Hence, dynophores extend classic 3D pharmacophores with statistical and sequential information about possible conformational states derived from MD simulations. In such a way, the static view of pharmacophores is transferred to a dynamic one with dynophores.

2 Methods

2.1 Pharmacophore Modelling

In CADD, the description of drug-target interactions with pharmacophores has evolved to a well-established method. A general definition for the term pharmacophore was published in 1998 by a IUPAC working party [18]:

> *A pharmacophore is the ensemble of steric and electronic features that is necessary to ensure the optimal supramolecular interactions with a specific biological target structure and to trigger (or to block) its biological response.*
>
> *A pharmacophore does not represent a real molecule or a real association of functional groups, but a purely abstract concept that accounts for the common molecular interaction capacities of a group of compounds towards their target structure. The pharmacophore can be considered as the largest common denominator shared by a set of active molecules.*

Hypothetical models such as pharmacophores must be (i) selective enough to filter active from inactive compounds, (ii) universal enough to find ligands as novel drug candidates, and (iii) suitable to automatically match large compound libraries. A common approach is to derive a model from distinct ligands in order to represent the specific mode of interaction by including or excluding descriptive chemical features. A classification of abstraction layers for chemical features is described in Table 1. By restricting general chemical feature definitions, the number of features increases at the cost of comparability. However, only comparable pharmacophores are sufficiently universal, and can represent a mode of action instead of a set of already existing ligands.

Layer	Classification	Example
4	chemical functionality without geometric constraint	ionisable area, lipophilic contact
3	chemical functionality with geometric constraint	hydrogen bond donor, acceptor
2	molecular graph descriptor without geometric constraint	atom, bond
1	molecular graph descriptor with geometric constraint	atom, bond

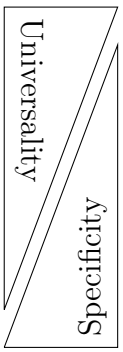


Table 1: **Classification of abstractions layers for chemical features**: the lower the abstraction level, the higher the specificity/selectivity and the lower the universality/ comparability of a chemical feature [19].

2.2 3D Pharmacophores in LigandScout

Using the pharmacophore modelling software LigandScout [20, 21], 3D pharmacophores can either be developed by (i) ligand-based drug design, starting from a set of ligands that are known to similarly bind to the same target, or by (ii) structure-based drug design, investigating specific ligand-target interactions if the complex structure is available.

In LigandScout, a molecular system is interpreted as *molecular environment* which is composed of (i) a *core molecule*, consisting of the ligand to be examined, and of (ii) a *macromolecule* consisting of the rest, i.e. the target, cofactors, water molecules, and ions. Structure-based 3D pharmacophores are generated from the interactions between core molecule and macromolecule.

2.2.1 Definition and Representation

3D pharmacophores in LigandScout consist of chemical features that are classified as abstraction layers 3 and 4 (Table 1). Pharmacophore features are visualised by point and vector features, respectively (Table 2): Point features are defined as 3D centre with scalar tolerance, and include hydrophobic (yellow sphere), aromatic (blue torus), positive (red star) and negative (blue star) ionisable areas. Hydrogen bond donors (green arrow) and acceptors (red arrow) belong to the vector features group [19].

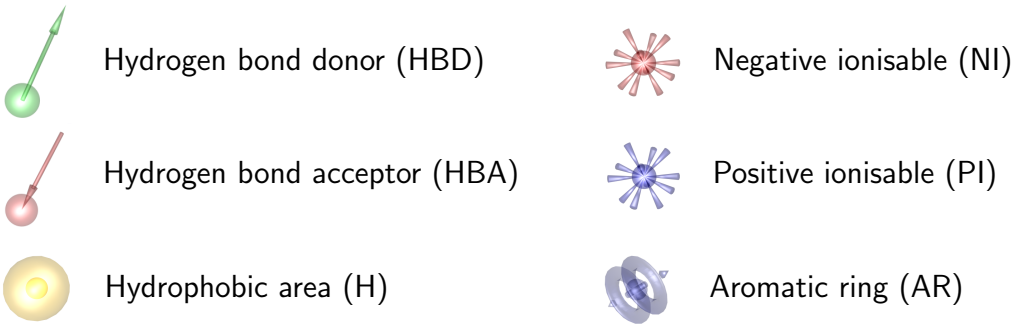


Figure 2: **Chemical feature types in LigandScout**: Point features include hydrophobic (yellow sphere), aromatic (blue torus), positive (red star) and negative (blue star) ionisable features. Vector features include hydrogen bond donors (green arrow) and acceptors (red arrow). By default feature tolerances range from 1.0 to 1.5 Å.

2.2.2 Data Structure

The ilib/LigandScout framework has been developed in the programming language Java. 3D pharmacophores are represented as `Pharmacophore` objects that contain all pharmacophore features as `ChemicalFeature3D` objects.

Pharmacophore class. The class `Pharmacophore` consists among others of a set of `ChemicalFeature3D` objects (`TreeSet<ChemicalFeature3D> features`) which represent

features such as hydrogen bond donors/acceptors or hydrophobic contacts. During pharmacophore generation, all potential features are generated separately on ligand- and target-site with a subsequent search for complementary features which will be saved as pharmacophore feature.

ChemicalFeature3D class. The class `ChemicalFeature3D` represents a pharmacophore feature, such as a hydrogen bond donor or acceptor, a hydrophobic area, or an aromatic ring. The `ChemicalFeature3D` class consists of

- the feature type (`String name`), such as a hydrogen bond donor,
- all atom serials on ligand-site that are involved in the feature (`Collection<Atom> involvedAtoms`),
- the coordinates for the geometric centre of the feature atoms (`double[] array3d`), and
- all atom serials on target-site that are involved in the feature (`Set<Atom> envAtoms`).

2.2.3 PML/PMZ File Format

The Pharmaceutical Markup Language (PML) is a XML-based file format storing information about ligand, target and pharmacophore in LigandScout. The compressed PML format is called PMZ. The PML format contains the tag <`MolecularEnvironment`>, describing the molecular environment in LigandScout, which is composed of

- <`CoreMolecule`>, information about atoms and bonds of the ligand,
- <`Environment`>, information about atoms and bonds of the ligand environment, consisting of the target, water and ions, and
- <`Pharmacophore`>, optional, the pharmacophore description, consisting of information about its features, such as feature type and involved atoms on ligand and environmental site.

2.3 Molecular Dynamics Simulations

Molecular Dynamics (MD) trajectories represent the possible evolution of a molecular system over time. Hence, MD simulations yield information on dynamics of a molecular system on an atomic level, not only in terms of protein dynamics and its configuration states but also in terms of target-ligand interactions.

2.3.1 Molecular Dynamics Force Fields

In MD, the possible evolution of a molecular system over time is calculated based on its force-field approximated potential energy. The principle of the MD trajectory calculation lies in the numerical integration of the Newtonian equations of motion. The force acting on a specific atom is calculated iteratively by taking the derivative of the potential energy function with respect to its position. The potential energy contains (i) covalent interactions, such as bond stretching, bond angle bending, and dihedral bending, and (ii) non-covalent interactions, such as van der Waals and Coulombic terms (Equation 1). The potential energy function $V(\mathbf{r}_{i,\dots,N})$ is described as function of the position $\mathbf{r}$ for N particles (Equation 2) [22].

$$V_{\text{total}} = \underbrace{\sum_{\text{bonds}} V_{\text{b}} + \sum_{\text{angles}} V_{\theta} + \sum_{\text{dihedrals}} V_{\phi}}_{V_{\text{covalent}}} + \underbrace{\sum_{i=1}^{N} \sum_{j=i+1}^{N} (V_{\text{vdW}} + V_{\text{Coulomb}})}_{V_{\text{non-covalent}}} \tag{1}$$

$$\begin{aligned} V(\mathbf{r}_{i,\dots,N}) = \sum_{\text{bonds}} \frac{k_i}{2}(l_i - l_{i,0})^2 + \sum_{\text{angles}} \frac{k_i}{2}(\theta_i - \theta_{i,0})^2 + \sum_{\text{dihedrals}} \frac{V_n}{2}(1 + \cos n\omega - \gamma) \\ + \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(4\epsilon_{ij}\left((\frac{\sigma_{ij}}{r_{ij}})^{12} - (\frac{\sigma_{ij}}{r_{ij}})^{6}\right) + \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \right) \end{aligned} \tag{2}$$

The potential energy function with its associated parameters is called force field.

2.3.2 DCD File Format

The DCD format is a binary FORTRAN file, containing the information about the MD trajectory for each saved time step of the simulation. This file format can be generated from any MD program output in e.g. VMD (Visual Molecular Dynamics) [23], an MD visualisation and analysis program.

3 Workflow: Dynophores

Structure-based 3D pharmacophores in LigandScout are generated from a static structure conformation, e.g. from the crystal structure or a snapshot of an MD simulation. Therefore, this pharmacophore only represents a static view on the interaction between target and ligand. *Dynophores* aim at extending classic 3D pharmacophores in LigandScout with statistical and sequential information about conformational states of a molecular system derived from MD simulations. The dynophore program *DynophoreApp* was implemented using the API of the ilib/LigandScout framework [20, 21]. The program automatically generates a pharmacophore for each snapshot of a given MD simulation and processes its features pursuing the following principle (Figure 3): Each feature is grouped into so-termed dynophore *superfeatures*. If a feature of the same type (e.g hydrogen bond donor or hydrophobic area) and the same involved atoms on ligand-site reoccurs over time, these features will build one group, the superfeature, to be monitored in its behaviour over the whole trajectory.

3.1 Dynophore Input

Dynophores include information on all LigandScout 3D pharmacophores generated for each frame of a MD trajectory. For a molecular system of interest, the dynophore generating program DynophoreApp requires (i) the assignment of all atoms to ligand- or target-site in LigandScout, and (ii) the MD trajectory to be analysed in terms of ligand-target-interactions as input (Figure 3).

LigandScout topology: PMZ format. For PMZ generation (PMZ definition in Section 2.2.3), the PDB structure of interest has to be loaded in the structure-based view of LigandScout, all atoms have to be assigned appropriately to the ligand (core molecule) or environmental (macromolecule) side, and proper bond interpretations have to be checked. The assignments can be saved as PMZ via File → Save as File.

MD trajectory: DCD format. The trajectory has to be prepared as follows before being saved as DCD (DCD definition in Section 2.3.2): (i) If the atoms of a molecular system are not all in one periodic box, but distributed over various boxes, they all have to be wrapped into a chosen image. In VMD, the `pbc wrap` function can be used to handle periodic boundary conditions. (ii) Since ligand movement and interaction with its environment are to be analysed, it is important for the dynophore visualisation that the trajectory is aligned on the environment of interest, e.g. the whole target molecule in the first trajectory frame. VMD provides a root mean square deviation (RMSD) alignment tool, accessible via VMD Main Menu → Extensions → Analysis → RMSD Trajectory Tool. This step is crucial for the spatial analysis of the graphical dynophore representation.

3.2 Dynophore Generation

DynophoreApp loads (i) the molecular environment from the PMZ file to define the core molecule (e.g. ligand), and macromolecule (e.g. target, cofactor, water or ions), and (ii) all trajectory frames from the DCD file.

For each frame t_i, the molecular environment is updated with the information given in the current frame from the MD trajectory, e.g. atom positions. For this configuration of the molecular system, a LigandScout 3D pharmacophore is generated. For each feature f_j of this pharmacophore, information about the interactions between ligand and its environment is saved in the following manner:

1. A *superfeature* is derived from feature f_j: The superfeature is defined by its (i) feature type and (ii) involved atoms on ligand-site. If this superfeature already occurred during the trajectory, information about feature f_j is added to it. If not, information about feature f_j is saved with the new superfeature. The superfeature, containing feature type and involved atoms, additionally gains the information about the frame it occurred in and the position of feature f_i. Visualised in 3D, features belonging to the group superfeature will therefore build a point cloud.

2. Information about the interaction between ligand and its environment is gathered. For feature f_j, all *interaction (=environmental) partners* on target-site are saved as follows: Each environmental partner is defined by its atoms interacting with the ligand. For each superfeature, all environmental partners over the whole trajectory are saved storing their occurrence or non-occurrence per frame. Additionally, the interaction distance is saved for all interactions over the trajectory for interaction quality assessment.

3.3 Dynophore Analysis Output

Generated information about dynophores (Section 3.2) is graphically represented and processed for feature and interaction pattern analysis as shown below:

Graphical representation of dynophores. Features grouped into one superfeature are saved in PML format for visualisation in LigandScout as 3D point clouds. Each point cloud represents one superfeature, and is clickable in LigandScout. Each point in a superfeature cloud represents one feature generated in one frame of the whole trajectory. The point cloud shapes depend on the alignment chosen before DCD generation. The superfeature point clouds can give an idea of the ligand-target dynamics over the trajectory: a very compact cloud can represent a strong ligand-target interaction, a cloud showing two different centres can point to distinct conformational changes during the trajectory, and an areal cloud can indicate unspecific ligand-target interactions. In order to concretise impressions from the spatial dynophore representation, statistical and sequential information on superfeatures, and their interaction patterns are available. An example is shown in Figure 3).

Statistical information. Superfeatures are statistically described by the frequency of their occurrence and by the frequencies of the occurrence of their interactions to the environment: how often a superfeature is formed and how often a superfeature interacts with which ligands can give rise to the importance of specific features and interaction patterns (examples are shown in Figure 7 on page 24). Additionally, interactions are statistically characterised by their distance, visualised as ligand-environment distance histograms. This descriptor can be used for estimating feature quality and relevance (an example is shown in Figure 10a on page 26).

Sequential information. Furthermore, all parameters used to gain statistical insights into ligand-target dynamics are also monitored over the trajectory: occurrences of superfeatures and their interaction partners (examples are shown in Figure 9 on page 25) as well as interaction distances (an example is shown in Figure 10b on page 26). Hence, conformational transitions and superfeature evolution as descriptor for ligand-target dynamics can be easily accessed.

Dynophores give a dynamic view on ligand-target interactions by including flexibility derived from MD simulations, and potentially enhance the predictive power of 3D pharmacophores. With the interplay of spatial, statistical and sequential information about ligand-target dynamics, dynophores provide a powerful analysis tool for the evaluation of pharmacophore features, interaction patterns and MD simulations.

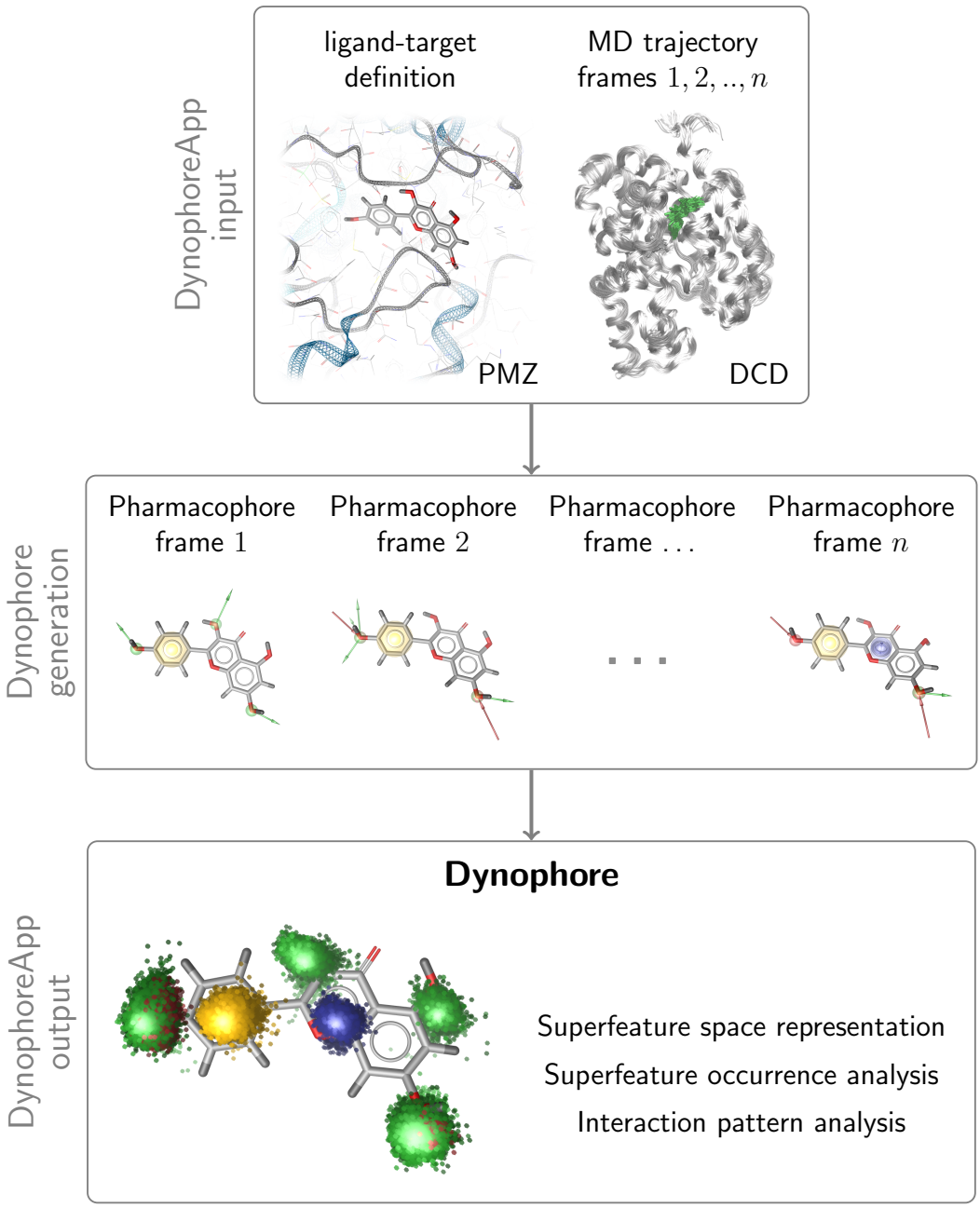


Figure 3: **Concept of dynophore generation with DynophoreApp**. *DynophoreApp input:* the user provides DynophoreApp with ligand-target definition and an MD trajectory of the molecular system to be analysed in PMZ and DCD format, respectively. *Dynophore generation:* DynophoreApp generates for each trajectory frame a LigandScout 3D pharmacophore and groups features into so-called superfeatures in terms of feature type and involved atoms on ligand-site. Additionally, all interaction partners per superfeature are saved. This dynamic pharmacophore is termed dynophore. *DynophoreApp output:* DynophoreApp outputs a graphical representation of all features grouped into superfeature as 3D point cloud each, and a statistical and sequential superfeature and interaction pattern analysis.

4 Implementation: DynophoreApp

In this study, DynophoreApp was implemented using the API of the ilib/LigandScout framework that has been developed in the programming language Java. The implementation of the DynophoreApp program is presented from dynophore generation to analysis.

DynophoreApp uses the LigandScout molecular environment structure which is composed of a core molecule and a macromolecule (Section 3.1). In the following, the core molecule is referred to by the term ligand and the macromolecule by the terms target or environment.

4.1 DynophoreApp: Dynophore Data Structure

Information about dynophores is stored in an object termed `Dynophore` whose data structure is presented in the following and summarised in Figure 4.

4.1.1 Dynophore Class

The `Dynophore` class contains ligand- and target-site related information for the detection and analysis of interaction patterns. The `Dynophore` class inherits from the `Pharmacophore` class (Section 2.2.2), and additionally contains

- a set of `SuperFeature` objects (`TreeSet<SuperFeature> superFeatures`) containing all dynophore superfeatures, as well as
- the number of MD trajectory frames (`int numFrames`).

4.1.2 SuperFeature Class

The `SuperFeature` class describes a superfeature that is defined by a certain feature type and by its involved atoms on ligand-site. Each superfeature is stored in one `SuperFeature` object. On the one hand, this class contains ligand-site related information about a specific superfeature, such as

- the superfeature type (`String featureName`), such as HBD, functioning as identifier key,
- all atom serial integers on ligand-site which are associated with this superfeature (`ArrayList<Integer> involvedAtomsSerials`), e.g. the oxygen serial of a hydroxyl group involved in a HBD feature or several serials in case of an AR feature, functioning as identifier key,
- a feature point cloud (`ChemicalFeatureCloud3D chemicalFeatureCloud3D`) grouping all features from the whole MD trajectory which belong to this superfeature, and

- the presence or absence of that superfeature in each trajectory frame as a boolean expression (`Boolean[] superFeatureEvents`).

On the other hand, the `SuperFeature` class also contains target-site related information. The interaction partners for a superfeature are monitored over the whole trajectory:

- each interaction (environmental) partner involved in this superfeature is stored in an `EnvPartner` object, and saved as set in the `SuperFeature` class (`Set<EnvPartner> envPartners`).

4.1.3 EnvPartner Class

The `EnvPartner` class describes one interaction partner on target-site over the whole trajectory. `EnvPartner` is defined by the atom serials on target-site which are involved in the superfeature. Each `EnvPartner` is stored in its associated `SuperFeature` object, and contains the following information:

- all atom serials on target-site which are involved in the interaction (`Set<Integer> envPartnerSerials`), functioning as identifier key,
- the name of the residue or component that contains the atom serials, e.g. LEU145 for a specific residue or PPS for a cofactor (`String envPartnerName`),
- the occurrence of the interaction leading to a feature in LigandScout as a boolean expression (`Boolean[] envPartnerEvents`), and
- the distances between the geometric centres of the superfeature building atoms on ligand-site and the interaction partner on target-site (`float[] envPartnerDistances`).

4.1.4 ChemicalFeatureCloud3D Class

The `ChemicalFeatureCloud3D` class describes all features over the whole trajectory which are grouped into one superfeature. It inherits from the `ChemicalFeature3D` class (see Section 2.2.2), and additionally contains

- the coordinates for the geometric centre of the feature cloud (`Vector3f cloudPosition`) and
- all points belonging to this feature cloud in a list of `AdditionalPoint` objects (`ArrayList<AdditionalPoint> additionalPoints`).

4.1.5 AdditionalPoint Class

The `AdditionalPoint` describes one feature that belongs to a certain pharmacophore generated in a certain trajectory frame, and that is grouped to a specific superfeature. This class therefore consists of

- the coordinates for a feature cloud point which represent the geometric centre of all atoms positions that are involved in this feature (`Vector3f pointPosition`), and
- the frame index to which the features belongs, in order to map each cloud point to the trajectory (`int frameIndex`).

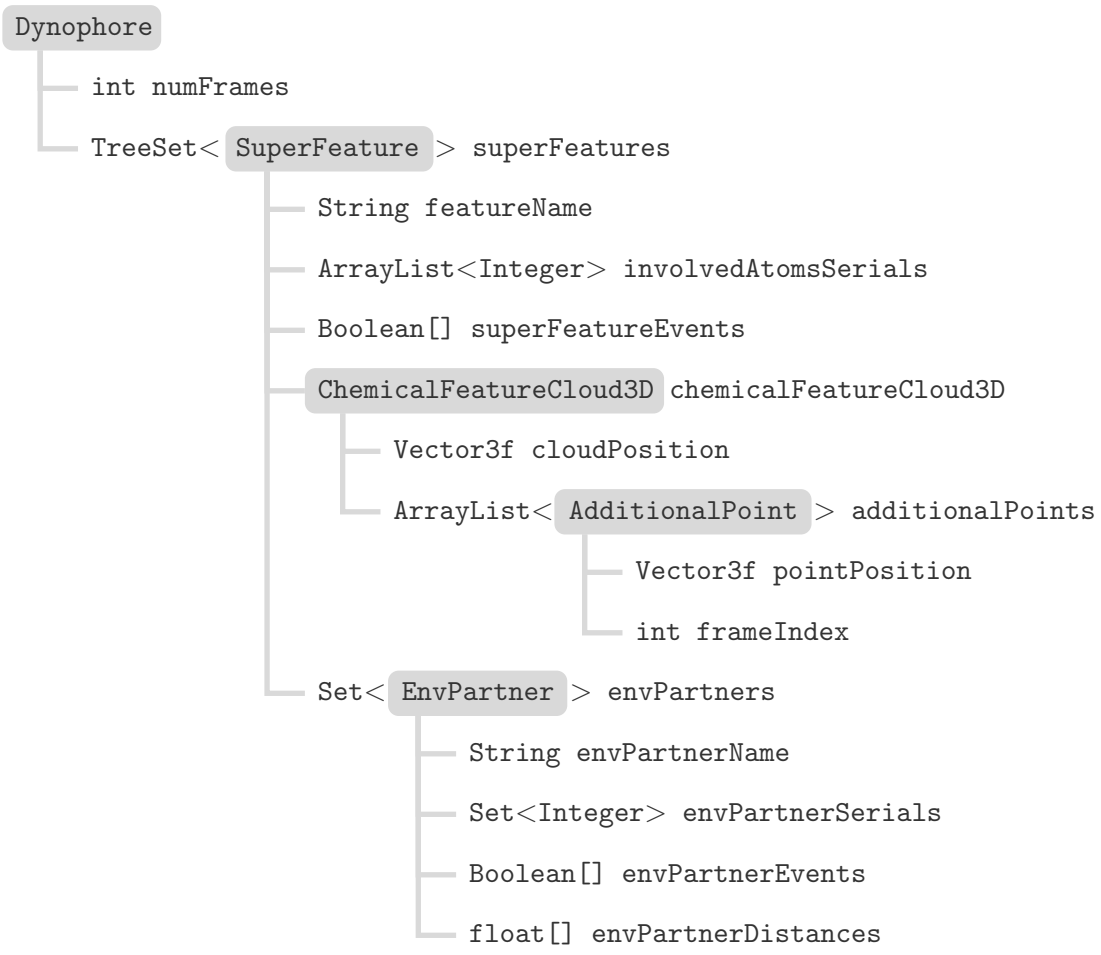


Figure 4: **Dynophore data structure**. DynophoreApp generates a dynophore in form of a Dynophore object which contains all superfeatures in SuperFeature objects. Each superfeature has a ChemicalFeatureCloud3D object that stores all associated features as point clouds in AdditionalPoint objects. Additionally, each superfeature encloses all interaction partners on target-site in EnvPartner objects. Grey boxes represent Java classes with their class fields listed below.

4.2 DynophoreApp: Program Workflow

The `DynophoreApp` subsequently invokes the classes `DynophoreGenerator` for dynophore input and generation, and `DynophoreAnalyser` for dynophore analysis and output. Using the method `ChemicalFeaturePreferencesChanger`, changes of the chemical feature preferences such as hydrogen bond distances and angles for hydrogen bonds can be set before dynophore generation and analysis.

4.2.1 DynophoreGenerator Class

`DynophoreGenerator` generates a `Dynophore` object from a PMZ and DCD input (Section 3.1). The `Dynophore` object is initialised, and the method `init()` reads the molecular environment of the system defining core molecule and macromolecule from the PMZ format as `MoleculeEnvironment` object, and all MD trajectory frames from the DCD format as `DCDTrajectory` object. The dynophore for that molecular system is generated by successively processing all trajectory frames with the method `processFrames()`: for each frame t_i, the molecular environment `MoleculeEnvironment` is updated with the current positions for all atoms of the system given in `DCDTrajectory`, and a structure-based 3D pharmacophore is created for this setting. The pharmacophore generating method is a core function of the ilib/LigandScout framework.

Subsequently, the method `processPharmacophore` iterates over each pharmacophore feature f_j, and invokes the `processFeature` method processing information on ligand and environmental site for each feature as follows:

1. The method `addFeatureToDynophore` generates a `SuperFeature` object from feature f_j that is defined by its feature type `featureName` and its involved atom serials on ligand-site `involvedAtomsSerials`. This `SuperFeature` object is compared to all `SuperFeature` objects that are already assigned to the superfeature set `TreeSet<SuperFeature> superFeatures` of the `Dynophore` object. (i) If this superfeature already exists, information about the frame it occurred in, and the feature f_j position is added to the `superFeatureEvents` and `ChemicalFeatureCloud3D` fields of the associated `SuperFeature` object. (ii) If the superfeature has not occurred before, it is added to the set of superfeatures with all corresponding information (Figure 4).

2. The method `addEnvPartnersToDynophore` considers all interaction partners of feature f_j. For each interaction partner on the environmental site of feature f_j, a `EnvPartner` object is generated defined by its interacting atoms `envPartnerSerials`. This `EnvPartner` is compared to all `EnvPartner` objects that are already in the set `Set<EnvPartner> envPartners` of the `Dynophore` object. (i) If the environmental partner is already in the set, information about the frame it occurred in is added to the `envPartnerEvents` field of the associated `EnvPartner`. (ii) If the environmental partner has not occurred before, it is added to the set of environmental partners with all corresponding information (Figure 4).

The method `processEnvPartnerDistances` is invoked after processing all frame pharmacophores with all their features. It iterates over all frames to save the distances of all superfeature feature points to their environmental partners to `envPartnerDistances` of the `EnvPartner` object.

4.2.2 DynophoreAnalyser Class

The `DynophoreAnalyser` covers the analysis of the generated dynophore as described in the following.

Dynophore visualisation. The method `writeDynophoreToPML` writes information about `ChemicalFeatureCloud3D` from the `Dynophore` object in PML format. This file format can be loaded in the structure-based view of LigandScout for a graphical representation of the dynophore. Examples: Figure 6, page 21.

Sequential superfeature and interaction occurrences. Occurrences of superfeatures and their interactions evolving over the MD trajectory (so-called *sequential* occurrence) are represented as multiple bar codes. These bar codes are plotted using the class `PlotBarCode` in the methods `plotSuperFeaturesBarCode` and `plotEnvPartnersBarCode`: The first method draws the occurrence of all dynophore superfeatures in a separate subplot. The second draws for each superfeature a plot that shows the occurrence of interactions with all its environmental partners in a separate subplot. Example: Figure 9, page 25.

Statistical superfeature and interaction occurrences. The frequencies of the occurrence of superfeatures and their interactions (so-called *statistical* occurrences) are saved as percentages.

Furthermore, interaction distance quality is sequentially and statistically visualised as distance series and histogram, respectively.

Sequential interaction distances. The method `plotEnvPartnersDistancesXYGraph` uses the `PlotXYGraph` class to create a plot for each superfeature showing all environmental partner in a trajectory-distance plot. Additionally, every distance data point that is associated with a superfeature is highlighted with a diamond-shaped marker. Since a feature is not only defined by its distance between ligand and environment, this marker provides additional information about the interaction type. Example: Figure 10b, page 26.

Statistical interaction distances. The method `plotEnvPartnersDistancesHistograms` creates for each superfeature a histogram from the `PlotHistogram` class that shows the distance distribution for each environmental partner in one plot. The distance distribution is only shown for those frames that provoke a feature. Example: Figure 10a, page 26.

Data for sequential plots are preprocessed by the `FramesDownsizer` class that downsizes the trajectory length to a user-defined length for appropriate graphical representation.

Since a trajectory of 20,000 data points cannot be resolved properly in a graph of $500x500$ pixel size, only 1,000 data points are used here. Data for bar code plots are merged, e.g. 20 frames to one, and assigned with occurrence (and frequency of occurrence) if half or more frames have an occurrence (Figure 9). Data for the trajectory-distance plot are not merged, but every e.g. 20th frame is used and represented with its information on distance and occurrence, (Figure 10b).

All plotting classes use the Java package JFreeChart [24]. The dynophore plots are currently saved as JPG files, and raw data from all `Dynophore` fields are saved as data files for individual further processing.

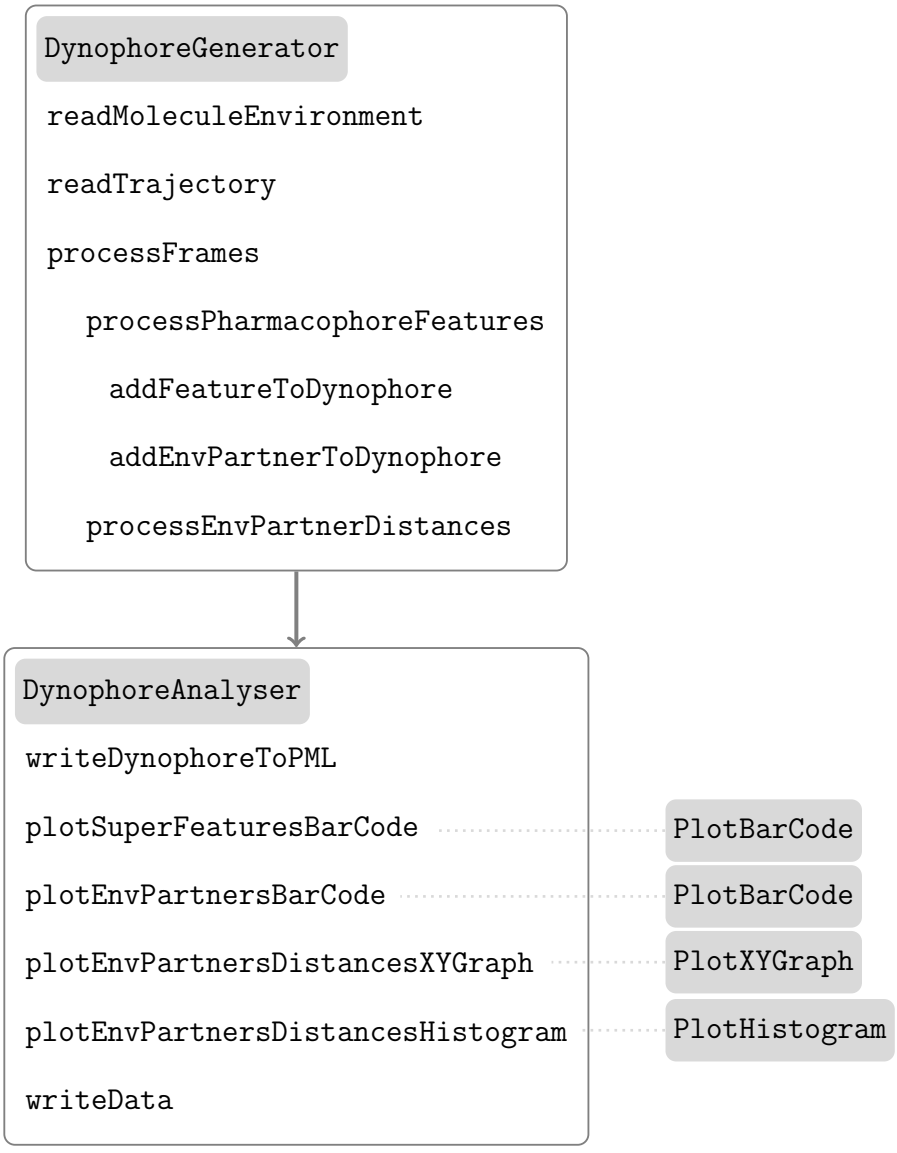


Figure 5: **DynophoreApp workflow**: Dynophore generating and analysing classes DynophoreGenerator and DynophoreAnalyser are presented with their main methods. Java classes are highlighted with grey boxes.

5 Applications

In this section, several applications for dynophores are demonstrated. Dynophore analyses can help the understanding of (i) ligand and feature dynamics, Section 5.1, (ii) ligand-environment interaction patterns during MD simulations, Section 5.2, (iii) MD-based feature importance in comparison with static pharmacophores, Section 5.3, and (iv) differences in ligand-target interactions of similar ligands, Section 5.4. DynophoreApp was tested on data from 100 ns MD simulations showing sulfotransferase SULT1E1 with cofactor phosphoadenosine-5'-phosphosulfonate (PAPS), and ligand kaempferol or 4-OH-2,3,5,2',4',5'-hexachlorobiphenyl (4-OH-HCB), Section 5.1 to 5.3. In Section 5.4, dynophores are discussed for 50 ns MD simulations with muscarinic M_2 acetylcholine receptor (M_2AChR) in complex with ligand iperoxo or isoxazol. The MD simulations were conducted by Christin Rakers and Marcel Bermudez, respectively, Institut für Pharmazie, Freie Universität Berlin.

Sulfotransferase SULT1E1. In phase II metabolism, sulfotransferases (SULT) regulate the sulfonation of endogenous and xenobiotic small molecules. The sulfonate group of SULT cofactor PAPS is transferred to the hydroxyl group of the substrate. SULTs are prone to inhibition by endo- or exogenous substances, leading to decreased sulfonation rates and therefore disruptions of the homeostasis of endogenous substances. Thus, the understanding of SULT is important for health risk assessment in drug discovery. Rakers et al. [17] presented a novel ligand prediction model for SULT1E1. MD simulations based on a crystal structure without ligand (PDB: 1HY3 [25]) were performed to sample protein conformational space. Ensemble docking with a database containing active ligands was used to generate ligand-protein docking poses. LigandScout 3D pharmacophores were generated for eight selected ligands in selected poses, enabling the identification of active ligands of SULT1E1 via virtual screening. In order to validate the pharmacophores, MD simulations were performed for the selected docking poses with three runs each. In the presented study, dynophores were generated and analysed for all MD simulations and are representatively discussed for ligands kaempferol and 4-OH-HCB.

Muscarinic M_2 acetylcholine receptor (M_2AChR). Muscarinic receptor M_2AChR belongs to the superfamily of G protein coupled receptors (GPCRs) mediating cell responses to messengers such as hormones, neurotransmitters, light, and physical stimuli. GPCRs have multiple integrative and highly dynamic signalling pathways including the activation of G proteins, β-arrestin and G protein interacting proteins (GIPs). Due to the broad ligand and signalling spectrum of GPCRs, a mechanistic understanding of ligand binding is important in drug discovery. Bock et al. [26] compared experimentally shown high differences in binding affinity for ligands iperoxo and isoxazol by performing and analysing MD simulations based on docking poses from active M_2AChR crystal structure (PDB: 4MQT [27]). In the presented study, dynophores were generated and analysed for both iperoxo and isoxazol MD simulations to help the understanding of ligand binding differences from a dynamically view.

5.1 Application: Dynophore Feature Space Analysis

The graphical representation of dynophores as 3D point clouds derived from MD simulations visualises ligand dynamics and feature distributions at a glance. The spatial distribution of dynophore features can give a first idea about (i) ligand flexibility and potential conformational states of the ligand within the active side, and (ii) ligand positioning, e.g. whether a ligand changes its initial MD position. Representative dynophores are discussion for MD simulations with SULT1E1 in complex with cofactor PAPS and kaempferol or 4-OH-HCB (Figure 6).

Dynophores for two different MD simulations with SULT1E1 in complex with PAPS and kaempferol show different feature distribution types. The dynophore derived from the first simulation is shown in Figure 6a. Well-shaped spheric feature point clouds with scarce outliers resemble the pharmacophore feature spheres in shape and diameter (Figure 2). Analysis of the associated MD trajectory shows stable ligand positioning in the docking pose used as simulation starting structure. However, a second simulation for the same ligand-target complex results in a differently shaped dynophore presented in Figure 6b. This dynophore shows an areal point distribution moving away from the initial ligand position, and several spheric point clouds slightly shifted away from the initial ligand position. Investigations on the associated MD simulation revealed that at the beginning of the simulation, the ligand slightly changes its starting position where it was positioned stably for 17,000 frames, represented in the dynophore with spheric point clouds. Afterwards, it moves away from the active side centre, indicated in the dynophore with the areal feature regions.

As a third example, the dynophore for an MD simulation of SULT1E1 in complex with PAPS and 4-OH-HCB is presented in Figure 6c. The ligand consists of two chloride containing phenyl rings connected by a single bond. One ring contains four chlorides and a hydroxyl group pointing to PAPS in the active side centre. The dynophore shows spheric point clouds for the HBD and HBA feature of the hydroxyl group and for four H features of the four chloride atoms. The dynophore for the second ring pointing towards the active side opening shows three H features for its three chloride ions as broad banana-shaped feature clouds. Investigations on the underlying MD simulation revealed that the ring pointing to the active side centre is rather inflexible, whereas the ring pointing to the active side opening is flexibly rotating along the single bond connecting both rings.

Thus, with the graphical representation of dynophores showing feature distributions in space, ligand flexibility as well as small or big changes in ligand positioning and interaction can be very easily accessed as a starting point for detailed and individual MD trajectory analysis.

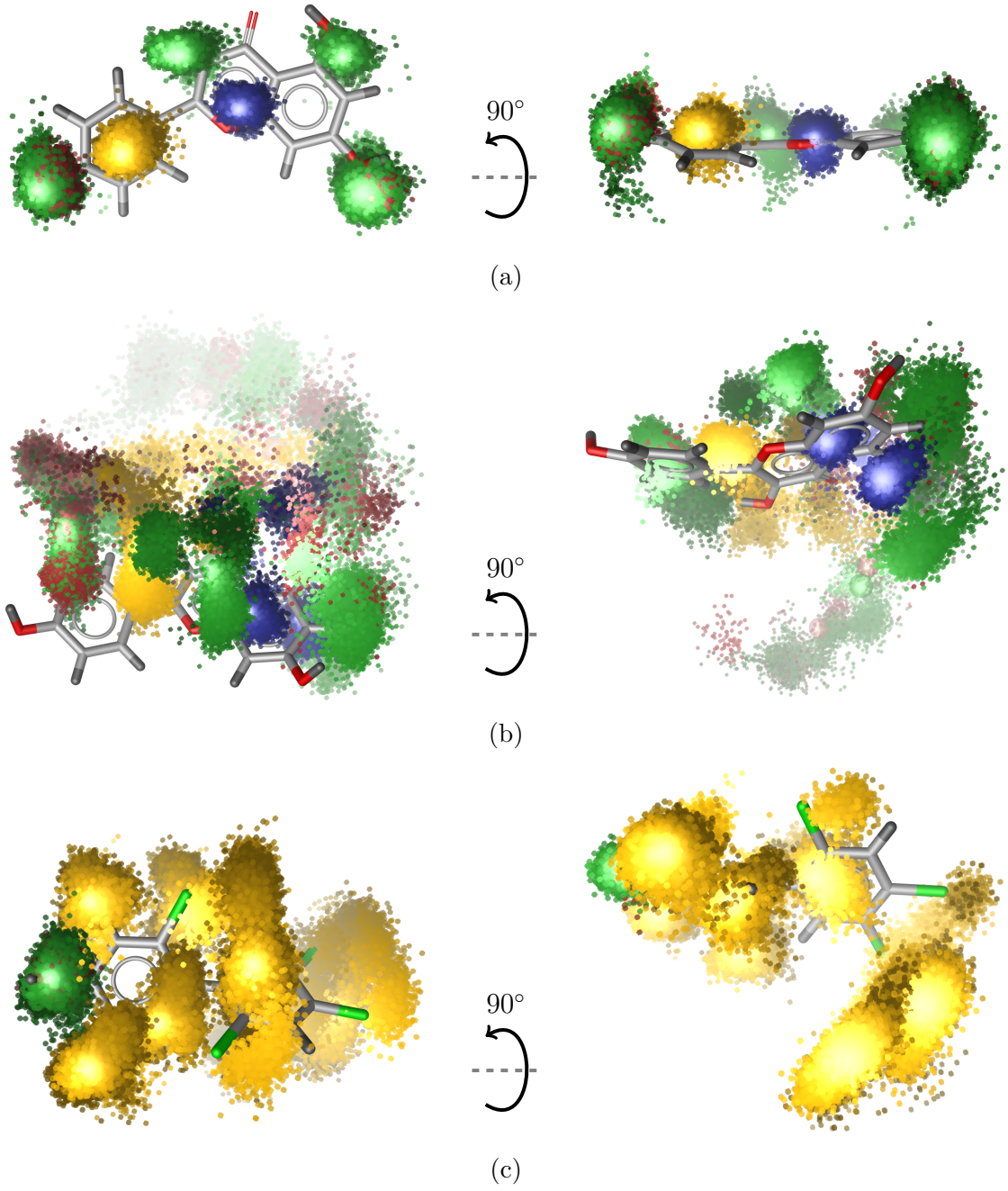


Figure 6: **Graphical representation of dynophores as spatial point clouds**. Dynophores derived from MD simulations of SULT1E1 in complex with PAPS and the following ligands: (a) rather inflexible kaempferol, (b) flexible kaempferol moving away from the active side, and (c) 4-OH-HCB having an inflexible and a rotating moiety towards the active side centre and opening, respectively. All dynophores are presented with ligands (conformation of first trajectory frame). All ligands are positioned with the active side centre containing cofactor PAPS to the left and the active side opening to the right.

5.2 Application: MD-based Interaction Pattern Analysis

DynophoreApp provides comprehensive information about the interaction pattern between ligand and its environment. The occurrence of a superfeature and the occurrence of its interactions to different environmental partners are described statistically as occurrence percentages, as well as sequentially as occurrence bar code plots. Furthermore, the interaction quality is described by interaction distances: statistics and trajectory monitoring for data on all interaction partners for one superfeature are assessable as distance histograms and trajectory-distance series, respectively.

The interaction pattern analysis is presented using SULT1E1 in complex with cofactor PAPS and kaempferol as an example. The interaction pattern in Figure 7 presents eight superfeatures different in their frequencies of occurrence, showing only those greater than 5%. Whereas the AR feature occurs only in 19% of the whole simulation, the hydroxyl group pointing to the active side centre occurs in 74% of the trajectory as HBD feature interacting with the catalytic His105 (65%) and the sulfonate oxygen atoms of the cofactor PAPS (31%, 30% and 12%). The associated ring has an H feature interacting over the whole trajectory (100%) with mainly aromatic amino acids such as phenylalanine and tyrosine. The ligand moiety pointing to the active side opening is among others stabilised by a hydroxyl group interacting with Asp20. Thus, the dynophore is in agreement with the reported interaction pattern at the catalytic centre of SULT involving the His105 induced transfer of the sulfonate group from PAPS to the catalytically important hydroxyl group of the ligand kaempferol [28].

Furthermore, the interactions of the hydroxyl group of kaempferol to the active side centre shown in Figure 8 were exemplary analysed in more detail. The sequential bar code plot shows an evenly distributed superfeature occurrence which is in 65% of the frames based on interactions with His105, and in 73% of the frames based on interactions with the PAPS sulfonate oxygen atoms (Figure 9). In the beginning of the MD simulation, the hydroxyl group of the ligand interacts with His105 (red) and one of the PAPS sulfonate oxygen atoms (green). A dynamic interplay of the ligand with His105 and the PAPS sulfonate group develops over the trajectory as shown in the interaction partner bar code plots: Single hydrogen bonds to either His105 or PAPS, and bifurcated hydrogen bonds to both are observed. The rotatable sulfonate group allows alternating interaction intervals for each of the three oxygen atoms with preference to two of them (green and cyan).

Interaction distance analyses give further insight into the interaction quality (Figure 10). The interaction distance histogram in Figure 10a shows the distribution for all frames that show a superfeature for the hydroxyl group of the ligand. The different percentages of occurrence discussed before result in different distribution peak areas. The distance distribution of His105 shows one sharp peak at hydrogen bonding distance around 2.8 Å. However, the distance distributions of all three PAPS sulfonate oxygen atoms show two peaks, one small peak at short hydrogen bonding distance of around 2.6 Å, and one large peak at larger distances. On the basis of distance analysis, ligand hydrogen bonding to His105 seems of higher quality (strength), whereas hydrogen bonding to PAPS mainly

shows weaker hydrogen bonds (higher distance peak) and only some stronger hydrogen bonds (lower distance peak). Hence, even though the amount of hydrogen bonds established to His105 and PAPS is similar, the interaction quality seems higher for His105.

Interaction distance trajectory series in Figure 10b shows all interaction distances over the trajectory with a diamond-shaped marker indicating superfeature occurrence in the marked trajectory frame. This plot can be useful to detect major changes in the interaction conformation. For instance, a large change of interaction distances or switches between two different distance values during the simulation indicates conformational changes. Findings like these represent a good starting point for further investigations of specific trajectory intervals and is therefore a helpful additional MD analysis tool to handle large MD trajectories. In the example shown here, only short minor changes in the interaction distances to PAPS are observed.

All in all, dynophores serve as a powerful MD analysis tool for the assessment of interaction patterns and their interaction quality. Statistics help to give an overview over superfeatures and their interactions. Sequential analysis provide access to trajectory clusters or unexpected events during the trajectory for further targeted analysis.

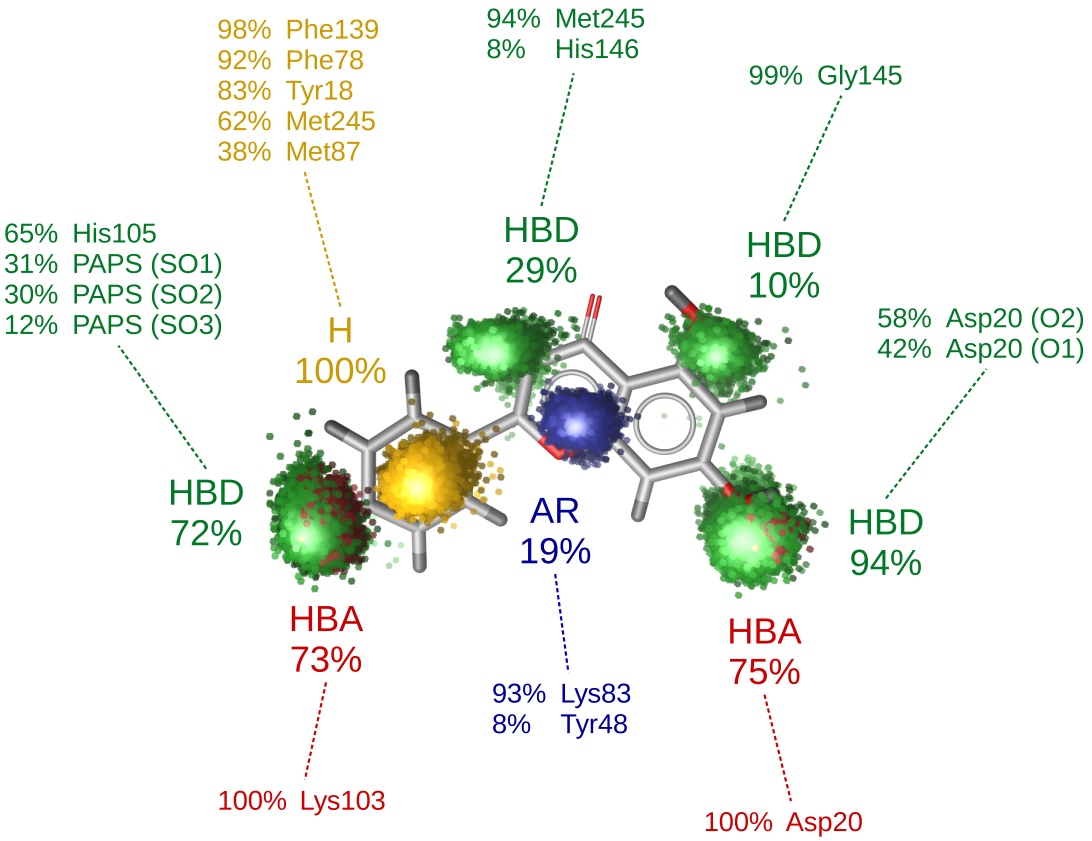


Figure 7: **Dynophore-based interaction pattern overview**. For an exemplary interaction pattern analysis, the dynophore for kaempferol bound to SULT1E1 and cofactor PAPS is presented showing all superfeatures with their environmental interacting partners. Occurrence statistics for the superfeatures and all interaction partners are listed, whereby only those with an occurrence higher than 5% are displayed. The percentages of occurrence are calculated for superfeatures on the basis of all frames, and for the interaction partners on the basis of all frames that show the specific superfeature.

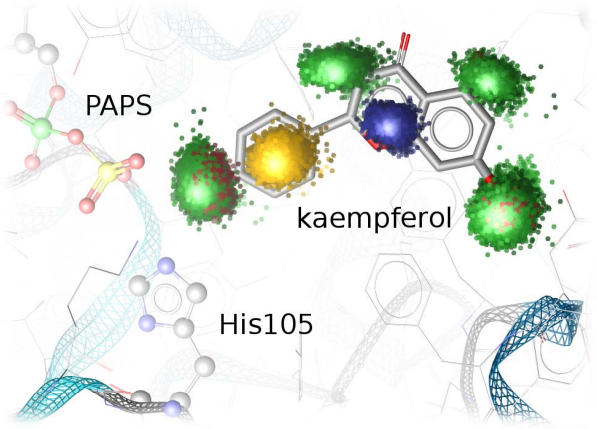


Figure 8: **SULT1E1 active side** with cofactor PAPS and ligand kaempferol. The superfeature HBD (green) at the catalytic hydroxyl oxygen of ligand kaempferol interacts with sulfonate oxygen atoms of cofactor PAPS and SULT1E1 His105. This interaction pattern is further analysed in Figure 9 and 10.

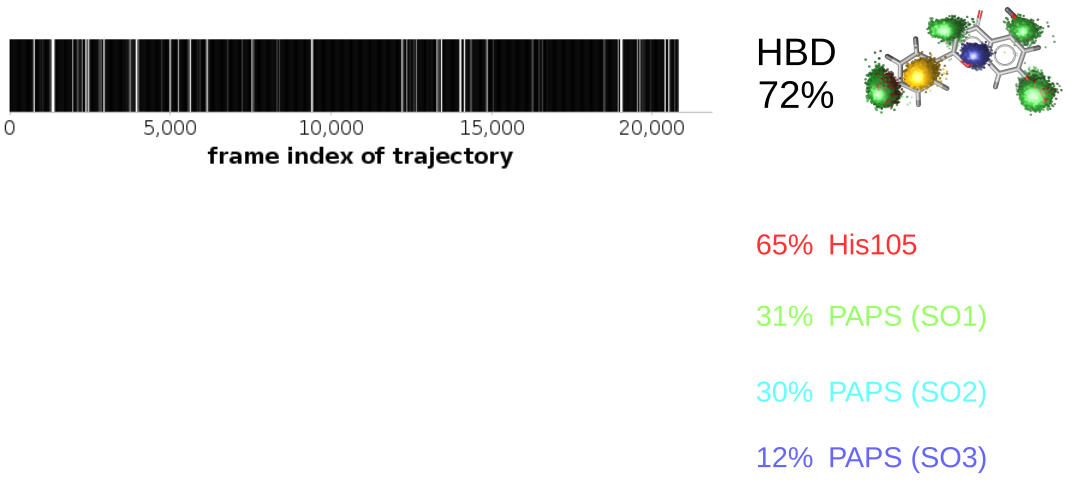
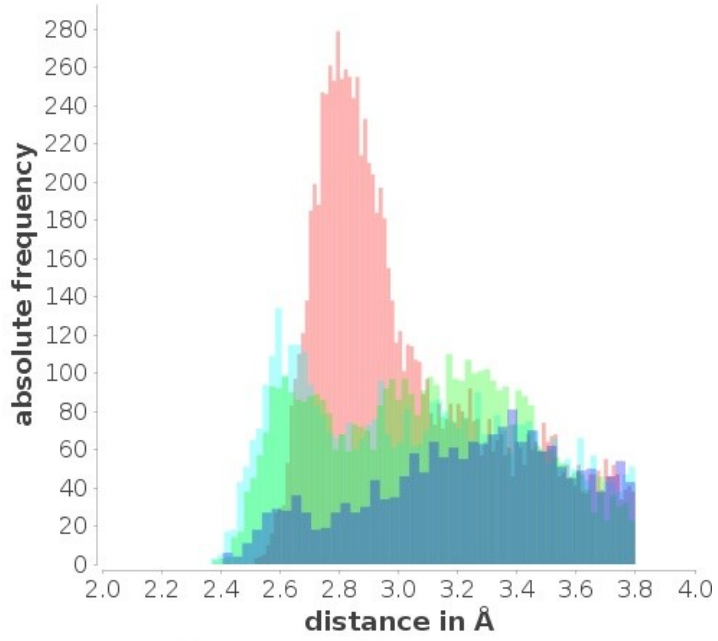
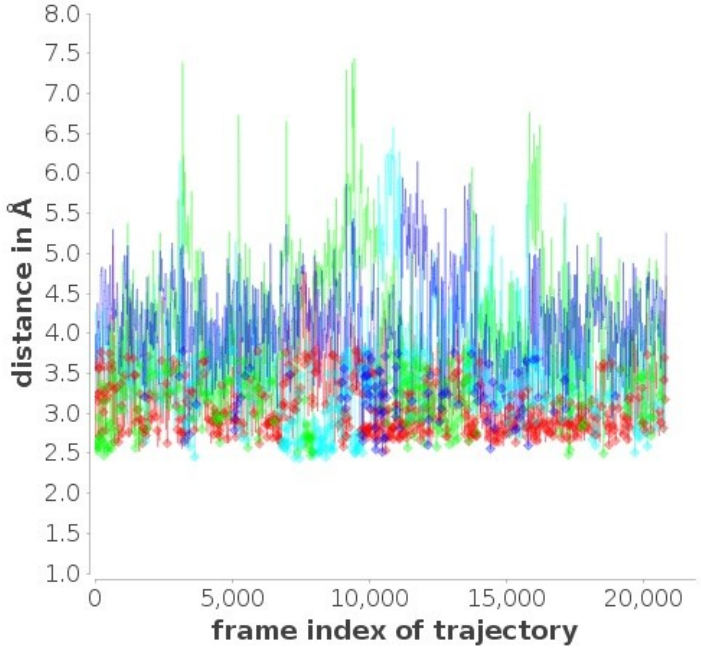


Figure 9: **Dynophore analysis: superfeature and interaction occurrences** are displayed in form of statistics (frequencies) and sequence over the trajectory (bar code plots). Interaction partners for the catalytic hydroxyl group of kaempferol (superfeature HBD) are His105 in red, and sulfonate oxygen atoms of PAPS in blue, green and cyan. See Figure 8 for structural arrangement of the active side.



(a)



(b)

Figure 10: **Dynophore analysis: interaction distances** in form of (a) a distance histogram for all frames with the superfeature present, as well as (b) a sequence over the trajectory with diamond-shaped markers at frames with superfeature present. Interaction partners for the catalytic hydroxyl group of kaempferol (superfeature HBD) are His105 in red, and sulfonate oxygen atoms of PAPS in blue, green and cyan. See Figure 8 for structural arrangement of the active side.

5.3 Application: Validation of Static Pharmacophores

Pharmacophores not only provide a compact view on ligand interactions, their 3D feature arrangement is also used for virtual screening to find structurally similar ligands that could bind to the same target. Pharmacophore features are optimised via virtual screening against a training set of active and inactive ligands. The same procedure is possible with dynophores but not yet implemented. However, dynophores already can be used to validate or optimise static pharmacophore models, since the additional information on dynamics can help to weight pharmacophore features. Statistics on the occurrence of a superfeature are referred to as frequency in the following.

The static pharmacophore shown in Figure 11a was derived from a default pharmacophore generation in LigandScout on the basis of a selected SULT1E1 docking pose with cofactor PAPS and ligand kaempferol, and a subsequent pharmacophore optimisation with virtual screening against a database of active and inactive ligands [17]. The optimised pharmacophore consists of five features, two HBD features, an H feature and two AR features. A dynophore is derived from an MD simulation of a docking pose with kaempferol.

The obtained dynophore (Figure 11b) resembles only those superfeatures assigned in the static pharmacophore. Their frequencies give an idea of the importance of each feature based on the underlying MD simulation. Both HBD features (72% and 94%) and the H feature (100%) are confirmed with high feature frequencies, whereas both AR features showed only low frequencies (19% and 1%). The dynophore in Figure 11c shows all superfeatures with a frequency higher than 5%, and reveals some additional superfeatures in contrast to the static pharmacophore. Both pharmacophore HBD features could also serve as HBA (73% and 75%), and two additional minor HBD features are observed with negligible occurrence (29% and 10%).

On the basis of these dynophore data, the static pharmacophore features could be discussed and revised if applicable. The frequencies of the dynophore features confirm on the one hand both HBD features and the H feature present in the static pharmacophore, but on the other hand suggest to omit both AR features. Furthermore, the dynophore suggests that both HBD features also serve as HBA features. Interaction pattern analysis shows that these two HBA features interacted with the Lys103 side chain and the amide of the Asp20 peptidyl bond, respectively. Compared to their HBD counterparts, which interact with the catalytic moieties PAPS and His105, and with the Asp20 side chain, respectively, the HBD features seem of higher importance than the HBA features. Hence, the dynophore analysis suggests the HBD features and the H feature present in the static pharmacophore as core features. For higher specificity, the 19% AR feature and both HBA features could be all or selectively included as mandatory or optional features. Various combinations of core and additional features could be tested via virtual screening against a database with active and inactive ligands to find the optimal feature combination which is highly depending on the biology of the target.

As a future prospect, dynophores themselves could be used for virtual screening using the feature frequency and distribution as matching criteria, read also Section 7.

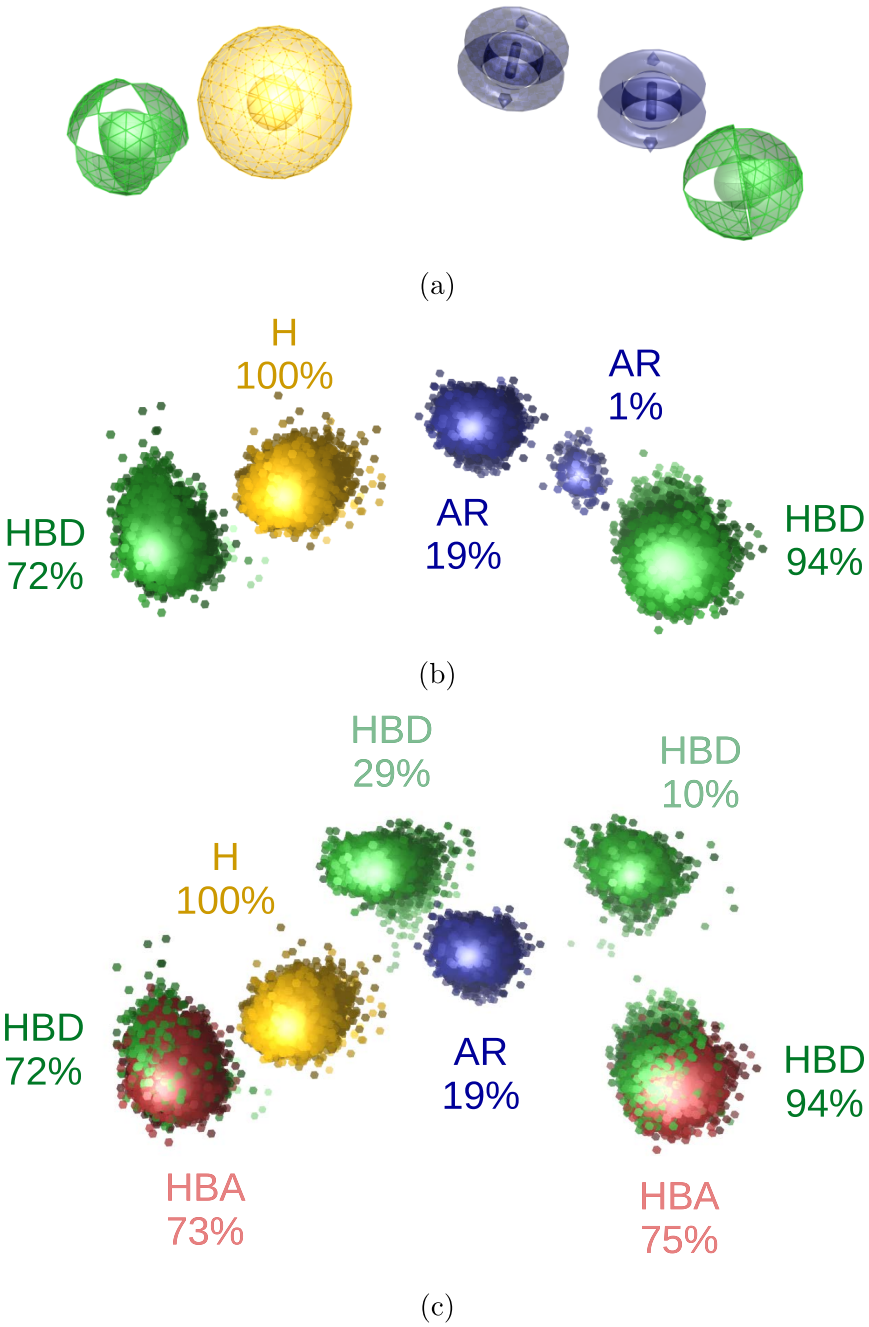


Figure 11: **Validation of static pharmacophores with dynophores**. (a) Static pharmacophore based on a docking pose for kaempferol bound to SULT1E1 and cofactor, and subsequent optimisation via virtual screening against a database with active and inactive ligands. (b) Dynophore with only those superfeatures shown that are also assigned in the static pharmacophore. (c) Dynophore with all superfeatures with a frequency higher than 5%. Frequencies of superfeature occurrences are annotated. Comparing static pharmacophore and dynophore, the AR features are under-represented in the MD simulation, and both HBD features are accompanied by HBA features with high weight.

5.4 Application: Comparative Feature Frequency Analysis

The ligand iperoxo shows high affinity for muscarinic receptors due to optimally positioned chemical features (Figure 12a, left): the hydrophobic triple bond stabilises the distance between the positively charged nitrogen and the 5-membered ring system containing an oxygen as HBA. However, the introduction of an aromatic ring system in isoxazol (Figure 12a, right) leads to a 100-fold lower binding affinity. Bock et al. [26] propose that the affinity difference can be explained by qualitative differences in the strength of hydrogen bonding between Asn404 and the oxygen in the ring systems, since furan-like systems like isoxazol build weaker hydrogen bonds [29]. Pharmacophore analysis shows no difference in ligand features and therefore can not explain the pharmacological data (Figure 12a). In order to access the difference dynamically, MD simulations for iperoxo and isoxazol bound systems were conducted and feature-wise analysed with dynophores due to hydrogen bond geometries between Asn404 and the ring oxygen of iperoxo and isoxazol, respectively.

Dynophores for the simulation of iperoxo and isoxazol bound to M_2AChRs are presented in Figure 12b. Spheric features confirm a stable ligand positioning with a favourable distance between the oxygen of the ring system (HBA feature) and the positively charged nitrogen (PI feature), linked via the triple bond (H feature). Furthermore, the H and PI features occur permanently during the simulations interacting with the same environmental partners. However, the simulations shows differences in the occurrence of the HBA feature at their ring oxygen: the dynophore analysis for iperoxo shows hydrogen bonding to Asn404 almost permanently in all frames (98%), whereas the interaction is formed only in 87% of all frames during the trajectory of isoxazol.

Dynophore-based sequential analysis shows that the HBA feature is evenly distributed over the whole trajectory for both iperoxo and isoxazol, in contrast to a permanent disruption of the feature during the trajectory. Thus, the lower frequency for HBA in case of isoxazol suggests a continuous lower hydrogen bond strength. Hence, further analysis on distances and angles characterising the hydrogen bond quality were conducted. Hydrogen bond distances are defined by the distance between the HBD and HBA atom, and hydrogen bond angles by the angle formed by the HBD, the hydrogen and the HBA atom. The stronger the hydrogen bond is, the shorter distances and the smaller (= more linear) angles exist [30]. The results show that iperoxo forms hydrogen bonds more often with shorter distances and smaller angles than observed in isoxazol, suggesting stronger hydrogen bonding in case of iperoxo (Figure 12c). The profound difference in affinity between iperoxo and isoxazol might at least partially be explained by the structural differences in hydrogen bonding of the oxygen in the rings systems as revealed by the dynophore analysis.

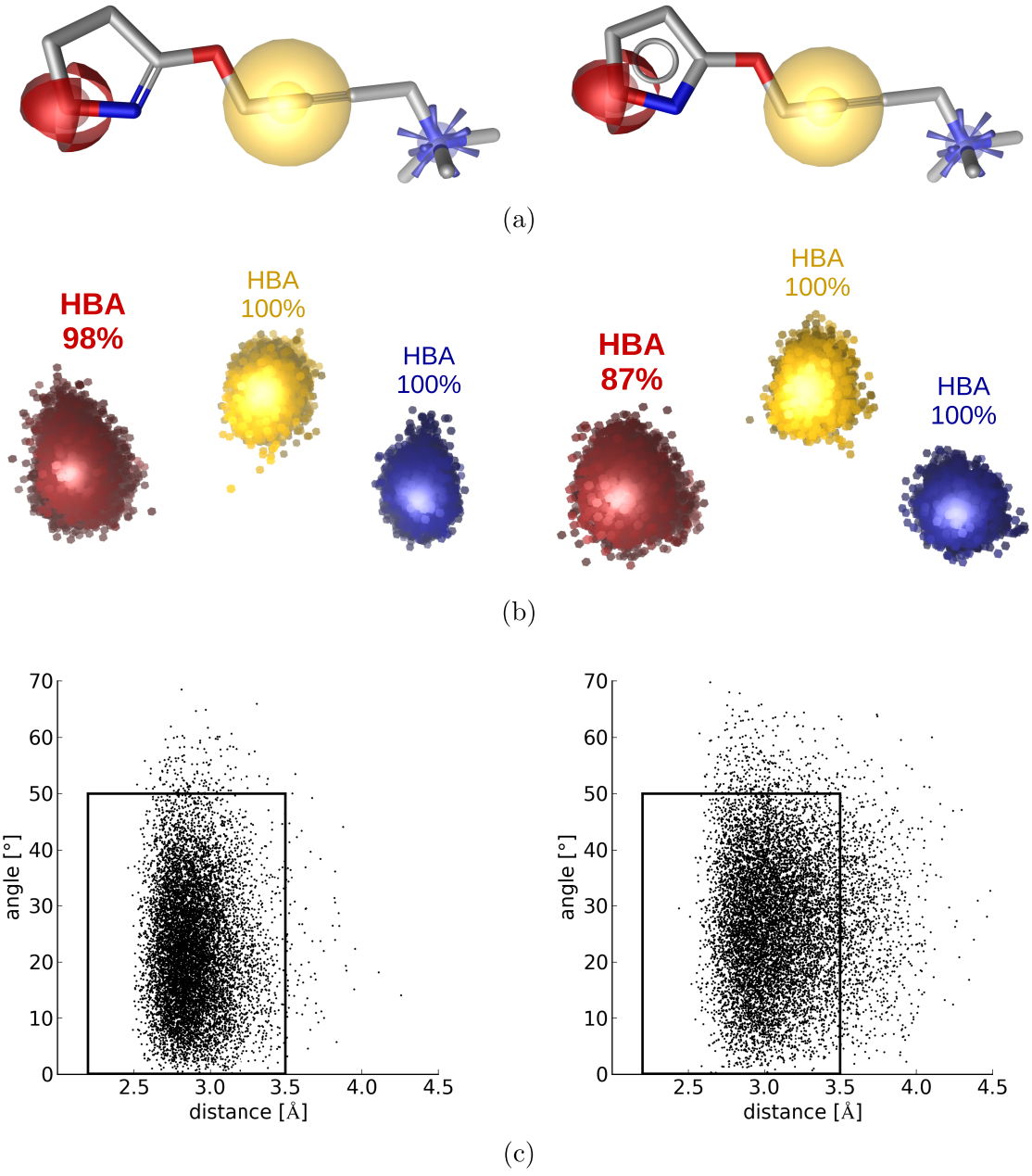


Figure 12: **Comparing feature frequencies of similar M_2AChR bound ligands iperoxo and isoxazol**. Key chemical interaction features of iperoxo (left column) and isoxazol (right column) bound to M_2AChR systems are presented as (a) static pharmacophores derived from the first trajectory frame of 50 ns MD simulations and (b) dynophores derived from the whole trajectory: iperoxo and isoxazol only differ in the HBA feature occurrence of 98% and 87%, respectively, possibly explaining the higher affinity of iperoxo to M_2AChR compared to isoxazol. (c) Distance and angle analysis of the hydrogen bond suggest a qualitatively stronger interaction for iperoxo than isoxazol. Hydrogen bond distances are defined by the distance between the HBD and HBA atom, and hydrogen bond angles by the angle formed by the HBD, the hydrogen and the HBA atom.

6 Discussion

Dynophores represent a novel approach of pharmacophore modelling that includes conformational flexibility of a molecular system derived from MD simulations without prior clustering or other preprocessing. Thus, dynophores represent the dynamics of a molecular system including all frames of the MD trajectory, and can therefore not only be applied to dynamic pharmacophore modelling but also to MD-based interaction pattern analysis.

In the following, limitations of the dynophore concept, and applications for the DynophoreApp analysis output are discussed.

6.1 Dynophore Limitations

When analysing and discussing dynophores, two main limitations must be considered: the DynophoreApp output is highly depending on the LigandScout feature definitions on the one hand, and on the MD simulation on the other hand.

LigandScout feature definitions. The dynophore, as any other pharmacophore approach, represents human-defined features that are thought to represent natural chemical properties. Whether the program sets a feature or not is defined by thresholds that are set carefully but artificially with LigandScout chemical feature definitions. Therefore, a sophisticated understanding of the feature definitions, and a careful examination of feature quality instead of relying only on the on/off information about features is crucial for the dynophore analysis. DynophoreApp provides an analysis of interaction distances for the assessment of interaction quality to give a first overview of the trajectory evolution. Further tools could be implemented such as interaction angle or binding energy analysis.

MD trajectory sampling. Dynophores aim to extend pharmacophore models with information about conformational space sampling derived from MD simulation. The quality of conformational space sampling in an MD trajectory determines the significance and statistical power of the associated dynophore. However, if a simulation shows a large population of one conformation, the conformation is not necessarily more important than others. For instance, the molecular system might simply be trapped in the energy landscape of that conformation, not able to sample other evenly or even more important conformational states. However, DynophoreApp will statistically analyse the trajectory and assign high weights to the superfeatures of that sampled conformation, suggesting high importance. Another simulation might relativise this statement due to a different conformational sampling. Many MD methods that more comprehensively assess conformational space are developed such as Markov state modelling or scaled MD [31, 32]. Therefore, it is important that the DynophoreApp user only utilises representative MD simulations for dynophore analysis, especially when arguing with dynophore statistics.

MD trajectory alignment. Dynophore superfeatures are graphically represented as spatial point clouds. The point cloud shapes are depending on the interaction dynamics of ligand and environment in the underlying MD simulation, see Section 5.1. However, the shape is also highly depending on the MD trajectory alignment conducted during the preprocessing of the DCD input file, see Section 2.3.2. Two different alignments, such as the alignment of the whole protein or of only the active side, will generate two different graphical representations of the dynophore. However, the statistical and sequential analysis is alignment-independent, since the analysis summarises all independently processed frames.

6.2 Dynophore Applications

Considering the discussed limitations and interpretation pitfalls, dynophores provide a widespread analysis tool for the assessment of conformational space and interaction pattern including spatial, statistical and sequential information. Potential applications for dynophores shown in this work include:

1. The graphical representation of dynophores showing feature distributions in space allow the easy assessment of ligand flexibility as well as small or big changes in ligand positioning and interaction. The results from such an analysis can be a starting point for detailed and individual MD trajectory analysis. (Section 5.1)

2. Dynophores serve as a powerful MD analysis tool for the investigation of interaction patterns and their interaction quality. DynophoreApp provides (i) statistics to give an overview on the distribution of superfeatures and interaction patterns, and (ii) analysis of the trajectory sequence to access trajectory clusters or unexpected events during the trajectory for further detailed analysis. (Section 5.2)

3. Dynophores can be used to validate or optimise static pharmacophore models, since the additional information on dynamics can help to weight pharmacophore features. (Section 5.3)

4. Furthermore, dynophore can give access to differences in ligand-target interactions for very similar ligands that show no binding differences with conventional static pharmacophores. (Section 5.4)

7 Conclusion and Future Prospects

The aim of this study was to develop pharmacophore models that include the conformational space of a molecular system derived from MD simulations. These dynamic pharmacophore models were implemented as dynophores in form of the DynophoreApp program using the API of the ilib/LigandScout framework. DynophoreApp is a comprehensive MD analysis tool with integrated pharmacophore definitions from LigandScout. The program groups features reoccurring over an MD trajectory into superfeatures, and provides an automated spatial, statistical and sequential analysis of their occurrence and interaction patterns.

Dynophores enrich classic static pharmacophores with information about the flexibility and conformational space of a molecular system. Therefore, dynophores potentially enhance the predictive power of pharmacophores and serve as helpful tool for the analysis of MD-based interaction patterns. Dynophore applications include (i) feature space analysis for the assessment of ligand-target flexibility, (ii) studies on MD-based interaction patterns, (iii) validation and improvement of static pharmacophores, as well as (iv) investigations on the importance of pharmacophore features for similar ligands.

Dynophores are applicable to diverse assignments at the intersection of pharmacophore modelling and MD simulation analysis. Hence, several future projects based on in this study implemented data structure for dynophores are of interest: (i) implementation of virtual screening with dynophores to potentially enhance virtual screening results, (ii) representation of the superfeature clouds as density function for further superfeature space analyses, and (iii) implementation of interaction angle analysis complementing the implemented interaction distance analysis for further assessment of interaction quality. Furthermore, dynophores are planned to be integrated more closely in the proprietary ilib/LigandScout software package and to be made publicly available.

A Appendix

LigandScout term	Definition
molecular environment	LigandScout interpretation of a system consisting of a core molecule, a macromolecule and optionally a pharmacophore
core molecule	molecular environment assigned as ligand in a structure-based approach
macromolecule	molecular environment assigned as target in a structure-based approach
HBA feature	hydrogen bond acceptor (3D pharmacophore feature)
HBD feature	hydrogen bond donor (3D pharmacophore feature)
H feature	hydrophobic area (3D pharmacophore feature)
AR feature	aromatic ring (3D pharmacophore feature)
PI feature	positive ionisable (3D pharmacophore feature)
NI feature	negative ionisable (3D pharmacophore feature)

Table A.1: **LigandScout terminology.** This table presents a short definition for main LigandScout terms.

Dynophore term	Definition
superfeature	all pharmacophore features that reoccur over the whole trajectory with the same feature type and the same involved atoms on ligand-site
dynophore	ensemble of all superfeatures for one core molecule
core molecule	molecule whose interaction properties are of interest, mostly a ligand, see Table A.1
environmental partner	molecular environment that is potentially interacting with the core molecule, including e.g. target, cofactors, water, ions
superfeature point cloud	spatial representation of all features over the trajectory grouped into one superfeature as points
spatial analysis	analysis of the superfeature point cloud distribution
statistical analysis	analysis of the frequency of a property during the whole trajectory
sequential analysis	analysis of the evolution of a property for the sequence of system conformations over the trajectory

Table A.2: **Dynophore terminology.** This table presents a short definition for dynophore terms that are introduced in this study.

References

[1] G Sliwoski, S Kothiwale, J Meiler, and E W Lowe. Computational methods in drug discovery. *Pharmacol. Rev.*, 66:334–95, 2014.

[2] J Mortier, C Rakers, M Bermudez, M S Murgueitio, S Riniker, and G Wolber. The impact of molecular dynamics on drug design: applications for the characterization of ligand-macromolecule complexes. *Drug Discov. Today*, 20:686–702, 2015.

[3] C M Stultz and M Karplus. MCSS functionality maps for a flexible protein. *Proteins*, 37:512–29, 1999.

[4] B Das, V Helms, V Lounnas, and R C Wade. Multicopy molecular dynamics simulations suggest how to reconcile crystallographic and product formation data for camphor enantiomers bound to cytochrome P-450cam. *J. Inorg. Biochem.*, 81:121–31, 2000.

[5] J-H Lin, A L Perryman, J R Schames, and J A McCammon. Computational drug design accommodating receptor flexibility: the relaxed complex scheme. *J. Am. Chem. Soc.*, 124:5632–3, 2002.

[6] J-H Lin, A L Perryman, J R Schames, and J A McCammon. The relaxed complex method: Accommodating receptor flexibility for drug design with an improved scoring scheme. *Biopolymers*, 68:47–62, 2003.

[7] J A McCammon. Target flexibility in molecular recognition. *Biochim. Biophys. Acta*, 1754:221–4, 2005.

[8] J L Knight and C L Brooks. Lambda-dynamics free energy simulation methods. *J. Comput. Chem.*, 30:1692–700, 2009.

[9] H A Carlson, K M Masukawa, K Rubins, F D Bushman, W L Jorgensen, R D Lins, J M Briggs, and J A McCammon. Developing a dynamic pharmacophore model for HIV-1 integrase. *J. Med. Chem.*, 43:2100–14, 2000.

[10] J Deng, K W Lee, T Sanchez, M Cui, N Neamati, and J M Briggs. Dynamic receptor-based pharmacophore model development and its application in designing novel HIV-1 integrase inhibitors. *J. Med. Chem.*, 48:1496–505, 2005.

[11] J Deng, T Sanchez, N Neamati, and J M Briggs. Dynamic pharmacophore model optimization: identification of novel HIV-1 integrase inhibitors. *J. Med. Chem.*, 49: 1684–92, 2006.

[12] Y-S Sohn, C Park, Y Lee, S Kim, S Thangapandian, Y Kim, H-H Kim, J-K Suh, and K W Lee. Multi-conformation dynamic pharmacophore modeling of the peroxisome proliferator-activated receptor γ for the discovery of novel agonists. *J. Mol. Graph. Model.*, 46:1–9, 2013.

[13] P B Burger, M Williams, J Sprenger, S B Reeksting, M Botha, I B Müller, F Joubert, L-M Birkholtz, and A I Louw. A novel inhibitor of Plasmodium falciparum spermidine synthase: a twist in the tail. *Malar. J.*, 14:54, 2015.

[14] N J Saez, M Mobli, M Bieri, I R Chassagnon, A K Malde, R Gamsjaeger, A E Mark, P R Gooley, La D Rash, and G F King. A dynamic pharmacophore drives the interaction between Psalmotoxin-1 and the putative drug target acid-sensing ion channel 1a. *Mol. Pharmacol.*, 80:796–808, 2011.

[15] A L Bowman and A Makriyannis. Approximating protein flexibility through dynamic pharmacophore models: application to fatty acid amide hydrolase (FAAH). *J. Chem. Inf. Model.*, 51:3247–53, 2011.

[16] K H Barakat, J Torin Huzil, T Luchko, L Jordheim, C Dumontet, and J Tuszynski. Characterization of an inhibitory dynamic pharmacophore for the ERCC1-XPA interaction using a combined molecular dynamics and virtual screening approach. *J. Mol. Graph. Model.*, 28:113–30, 2009.

[17] C Rakers, F Schumacher, W Meinl, H Glatt, B Kleuser, and G Wolber. In silico prediction of human sulfotransferase 1E1 activity guided by pharmacophores from molecular dynamics simulations. *J. Biol. Chem.*, 2015.

[18] C G Wermuth, C R Ganellin, P Lindberg, and L A Mitscher. Glossary of terms used in medicinal chemistry (IUPAC Recommendations 1998). *Pure Appl. Chem.*, 70:1129–43, 1998.

[19] G Wolber and R Kosara. Pharmacophores from Macromolecular Complexes with LigandScout. In L Thierry and R D Hoffmann, editors, *Pharmacophores and Pharmacophore Searches*, chapter 6, pages 131–50. Wiley-VCH, volume 32 edition, 2006.

[20] G Wolber and T Langer. LigandScout: 3-D pharmacophores derived from protein-bound ligands and their use as virtual screening filters. *J. Chem. Inf. Model.*, 45: 160–9, 2005.

[21] G Wolber, A A Dornhofer, and T Langer. Efficient overlay of small organic molecules using 3D pharmacophores. *J. Comput. Aided. Mol. Des.*, 20:773–88, 2006.

[22] A R Leach. Molecular Dynamics Simulation Method. In *Mol. Model. Princ. Appl.*, chapter 7. Pearson Education Limited, 2001.

[23] W Humphrey, A Dalke, and K Schulten. VMD: visual molecular dynamics. *J. Mol. Graph.*, 14:33–8, 27–8, 1996.

[24] D Gilbert. JFreeChart, www.jfree.org/jfreechart, 2015.

[25] L C Pedersen, E Petrotchenko, S Shevtsov, and M Negishi. Crystal structure of the human estrogen sulfotransferase-PAPS complex: evidence for catalytic role of Ser137 in the sulfuryl transfer reaction. *J. Biol. Chem.*, 277:17928–32, 2002.

[26] A Bock, M Bermudez, B Chirinda, M De Amici, U Holzgrabe, F Krebs, M Lohse, D Sydow, and G Wolber. Multidirectional binding control of G protein coupled receptor dynamics as a basis for agonist efficacy. *manuskript in preparation*, 2015.

[27] A C Kruse, A M Ring, A Manglik, J Hu, K Hu, K Eitel, H Hübner, E Pardon, C Valant, P M Sexton, A Christopoulos, C C Felder, P Gmeiner, J Steyaert, W I Weis, K C Garcia, J Wess, and B K Kobilka. Activation and allosteric modulation of a muscarinic acetylcholine receptor. *Nature*, 504:101–6, 2013.

[28] Z E Tibbs, K J Rohn-Glowacki, F Crittenden, A L Guidry, and C N Falany. Structural plasticity in the human cytosolic sulfotransferase dimer and its role in substrate selectivity and catalysis. *Drug Metab. Pharmacokinet.*, 30:3–20, 2015.

[29] C Bissantz, B Kuhn, and M Stahl. A medicinal chemist's guide to molecular interactions. *J Med Chem*, 53:5061–84, 2010.

[30] T Steiner. The hydrogen bond in solid state. *Angew. Chem. Int. Ed.*, 41:48–76, 2002.

[31] J D Chodera and F Noé. Markov state models of biomolecular conformational dynamics. *Curr. Opin. Struct. Biol.*, 25:135–44, 2014.

[32] W Sinko, Y Miao, C A F de Oliveira, and J A McCammon. Population based reweighting of scaled molecular dynamics. *J. Phys. Chem. B*, 117:12759–68, 2013.